#include<iostream>

#include<fstream>

#include <sstream>

using namespace std;

struct Node

{

string name;

long long int phone\_number;

Node \*next;

Node \*prev;

};

class ContactBook

{

Node \*head;

string x;

long long int y;

public:

ContactBook()

{

head=NULL;

x="";

y=0;

}

void CreateNode()

{

if(head==NULL)

{

Node \*newer= new Node;

cout<<" Name of Contact: ";

cin>>x;

newer->name=x;

cout<<" Phone Number: ";

cin>>y;

newer->phone\_number=y;

newer->next=NULL;

newer->prev==NULL;

head=newer;

cout<<" Contact Added"<<endl;

}

else

{

Node \*newer= new Node;

cout<<" Name of Contact: ";

cin>>x;

newer->name=x;

cout<<" Phone Number: ";

cin>>y;

newer->phone\_number=y;

newer->next=NULL;

Node \*temp=head;

while(temp->next!=NULL)

{

temp=temp->next;

}

temp->next=newer;

newer->prev=temp;

cout<<" Contact Added"<<endl;

}

}

void Display()

{

Node \*temp=head;

int count=0;

if(temp==NULL)

{

cout<<" No Contacts... Please Add Some Contacts"<<endl;

cout<<" Thanks"<<endl;

}

else

{

BubbleSort();

cout<<" Name: "<<" Number: \n"<<endl;

while(temp!=NULL)

{

count++;

cout<<" "<<temp->name;

cout<<" "<<temp->phone\_number<<endl;

temp=temp->next;

}

cout<<" Total contacts: "<<count<<endl;

}

}

void Search()

{

bool check=false;

Node \*temp=head;

cout<<"\*\*\*\*\*"<<endl;

cout<<" Press 1 if you want to Search By Name."<<endl;

cout<<" Press 2 if you want to Search By Number."<<endl;

int command;

cout<<" Enter the Command: ";

cin>>command;

if(command==1 && temp!=NULL)

{

cout<<" Enter the Name to Search: ";

cin>>x;

while(temp!=NULL)

{

if(temp->name==x)

{

cout<<"\*\*\*\*\*"<<endl;

cout<<" Name: "<<temp->name<<endl;

cout<<" Phone Number:"<<temp->phone\_number<<endl;

cout<<"\*\*\*\*\*"<<endl;

check=true;

break;

}

temp=temp->next;

}

if(check==false)

{

cout<<" Name Not Found"<<endl;

}

}

else if(command==2)

{

cout<<" Enter the Number to Search: ";

cin>>y;

while(temp!=NULL)

{

if(temp->phone\_number==y)

{

cout<<"\*\*\*\*\*"<<endl;

cout<<" Name: "<<temp->name<<endl;

cout<<" Phone Number: "<<temp->phone\_number<<endl;

cout<<"\*\*\*\*\*"<<endl;

check=true;

break;

}

temp=temp->next;

}

if(check==false)

{

cout<<" Number Not Found"<<endl;

}

}

}

void DeleteAllContacts()

{

Node \*temp=head;

Node \*temp2;

if(head==NULL)

{

cout<<" Already Contact Book is Empty"<<endl;

cout<<"\*\*\*\*\*"<<endl;

}

else

{

while(temp!=NULL)

{

temp2=temp;

temp=temp->next;

delete temp2;

}

head=NULL;

cout<<" Successfully Deleted All Contacts"<<endl;

cout<<"\*\*\*\*\*"<<endl;

}

}

void DeleteContactBySearch()

{

Node \*temp=head;

cout<<"\*\*\*\*\*"<<endl;

cout<<" Press 1 if you want to Search By name"<<endl;

cout<<" Press 2 if you want to Search By Number"<<endl;

int command;

cout<<" Enter the Command: ";

cin>>command;

if(command==1)

{

bool Dcheck=false;

cout<<" Enter the Name to Delete: ";

cin>>x;

while(temp!=NULL)

{

if(temp->name==x)

{

cout<<"\*\*\*\*\*"<<endl;

cout<<" Name: "<<temp->name<<endl;

cout<<" Phone Number: "<<temp->phone\_number<<endl;

cout<<"\*\*\*\*\*"<<endl;

Dcheck=true;

break;

}

temp=temp->next;

}

if(Dcheck==true)

{

int command;

cout<<" Press 1 to Delete the Contact: ";

cin>>command;

if(command==1 & temp==head)

{

Node \*temp1;

temp1=temp;

temp=temp->next;

delete temp1;

temp->prev=NULL;

head=temp;

cout<<" Contact Deleted Success Fully"<<endl;

}

else if(command==1 & temp->next==NULL)

{

temp->prev->next=NULL;

delete temp;

cout<<" Contact Deleted Success Fully"<<endl;

}

else if(command==1)

{

Node \*temp1;

temp1=temp;

temp->prev->next=temp1->next;

temp->next->prev=temp1->prev;

delete temp1;

cout<<" Contact Deleted Success Fully"<<endl;

}

else

{

cout<<" You Enter Wrong Command ... Try Again"<<endl;

}

}

else if(Dcheck==false)

{

cout<<" Contact of This Name Not Found."<<endl;

}

}

else if(command==2)

{

bool Dcheck=false;

cout<<" Enter the Number to Delete: ";

cin>>y;

while(temp!=NULL)

{

if(temp->phone\_number==y)

{

cout<<"\*\*\*\*\*"<<endl;

cout<<"name: "<<temp->name<<endl;

cout<<"Phone Number: "<<temp->phone\_number<<endl;

cout<<"\*\*\*\*\*"<<endl;

Dcheck=true;

break;

}

temp=temp->next;

}

if(Dcheck==true)

{

int command;

cout<<" Press 1 to Delete the Contact: ";

cin>>command;

if(command==1 & temp==head)

{

Node \*temp1;

temp1=temp;

temp=temp->next;

delete temp1;

temp->prev=NULL;

head=temp;

cout<<" Contact Deleted Success Fully"<<endl;

}

else if(command==1 & temp->next==NULL)

{

temp->prev->next=NULL;

delete temp;

cout<<" Contact Deleted Success Fully"<<endl;

}

else if(command==1)

{

Node \*temp1;

temp1=temp;

temp->prev->next=temp1->next;

temp->next->prev=temp1->prev;

delete temp1;

cout<<" Contact Deleted Success Fully"<<endl;

}

else

{

cout<<" You Enter Wrong Command ... Try Again"<<endl;

}

}

else if(Dcheck==false)

{

cout<<" Contact of This Name Not Found."<<endl;

}

}

else

{

cout<<" You Enter wrong Command"<<endl;

}

}

void BubbleSort()

{

Node \*temp=head;

Node \*i,\*j;

string n;

long long int n2;

if(temp=NULL)

{

cout<<" Empty contact Book"<<endl;

}

else

{

for(i=head;i->next!=NULL;i=i->next)

{

for(j=i->next;j!=NULL;j=j->next)

{

if(i->name>j->name)

{

n=i->name;

i->name=j->name;

j->name=n;

n2=i->phone\_number;

i->phone\_number=j->phone\_number;

j->phone\_number=n2;

}

}

}

}

}

void EditContacts()

{

Node \*temp=head;

cout<<"\*\*\*\*\*"<<endl;

cout<<" Press 1 if you want to Search By Name"<<endl;

cout<<" Press 2 if you want to Search By Number"<<endl;

int Ecommand;

cout<<" Enter the Command: ";

cin>>Ecommand;

if(Ecommand==1)

{

bool Echeck=false;

cout<<" Enter the Name to Edit: ";

cin>>x;

while(temp!=NULL)

{

if(temp->name==x)

{

cout<<"\*\*\*\*\*"<<endl;

cout<<"Name: "<<temp->name<<endl;

cout<<"Phone Number: "<<temp->phone\_number<<endl;

cout<<"\*\*\*\*\*"<<endl;

Echeck=true;

break;

}

temp=temp->next;

}

if(Echeck==true)

{

int command;

cout<<" Press 1 to Edit the Contact: ";

cin>>command;

if(command==1)

{

cout<<" Enter New Name: ";

cin>>x;

cout<<" Enter New Number: ";

cin>>y;

temp->name=x;

temp->phone\_number=y;

cout<<" Contact Edited Success Fully"<<endl;

}

else

{

cout<<" You Enter Wrong Command ... Try Again"<<endl;

}

}

else if(Echeck==false)

{

cout<<" Contact Not Found"<<endl;

}

}

else if(Ecommand==2)

{

bool Echeck=false;

cout<<" Enter the Number to Edit: ";

cin>>y;

while(temp!=NULL)

{

if(temp->phone\_number==y)

{

cout<<"\*\*\*\*\*"<<endl;

cout<<"Name: "<<temp->name<<endl;

cout<<"Phone Number: "<<temp->phone\_number<<endl;

cout<<"\*\*\*\*\*"<<endl;

Echeck=true;

break;

}

temp=temp->next;

}

if(Echeck==true)

{

int command;

cout<<" Press 1 to Edit the Contact: ";

cin>>command;

if(command==1)

{

cout<<" Enter New Name: ";

cin>>x;

cout<<" Enter New Number: ";

cin>>y;

temp->name=x;

temp->phone\_number=y;

cout<<" Contact Edited Success Fully"<<endl;

}

else

{

cout<<" You Enter Wrong Command"<<endl;

}

}

else if(Echeck==false)

{

cout<<" There is No Contact of this Number."<<endl;

}

}

else

{

cout<<" You Enter Wrong Command ... Try Again"<<endl;

}

}

void OflineSave()

{

Node \*temp=head;

ofstream myfile ("contactbook.txt");

if (myfile.is\_open())

{

while(temp!=NULL)

{

myfile<<temp->name<<endl;

myfile<<temp->phone\_number<<endl;

temp=temp->next;

}

myfile.close();

Structure();

}

else

{

cout<<" Thanks file is empty."<<endl;

}

}

void reopenCB()

{

bool isEmpty;

ifstream myfile ("contactbook.txt");

if (myfile.is\_open() & myfile.peek() != EOF)

{

int i=0;

while(getline(myfile,x))

{

if(i % 2 == 0)

{

if(head==NULL)

{

Node \*newer= new Node;

newer->name=x;

newer->next=NULL;

newer->prev==NULL;

head=newer;

}

else

{

Node \*newer= new Node;

newer->name=x;

newer->next=NULL;

Node \*temp=head;

while(temp->next!=NULL)

{

temp=temp->next;

}

temp->next=newer;

newer->prev=temp;

}

}

else

{

Node \*temp=head;

if(temp->phone\_number==0)

{

stringstream convert(x);

convert>>y;

temp->phone\_number=y;

}

else

{

Node \*temp=head;

while(temp->next!=NULL)

{

temp=temp->next;

}

stringstream convert(x);

convert>>y;

temp->phone\_number=y;

}

}

i++;

}

myfile.close();

}

else

{

cout<<" File is Empty so Cannot open...Sorry"<<endl;

}

}

void Structure()

{

cout<<"\*\*\*\*\*"<<endl;

cout<<" 1. Add Contact"<<endl;

cout<<" 2. Edit the Contact"<<endl;

cout<<" 3. Delete Contact"<<endl;

cout<<" 4. Search Contact"<<endl;

cout<<" 5. Display All Contacts"<<endl;

cout<<" 6. Delete All Contacts"<<endl;

cout<<"\*\*\*\*\*"<<endl;

int Scommand;

cout<<" Enter the Command: ";

cin>>Scommand;

try

{

if(Scommand>=1&&Scommand<=6)

{

if(Scommand==1)

{

CreateNode();

OflineSave();

Structure();

}

else if(Scommand==2)

{

EditContacts();

OflineSave();

Structure();

}

else if(Scommand==3)

{

DeleteContactBySearch();

Structure();

}

else if(Scommand==4)

{

Search();

Structure();

}

else if(Scommand==5)

{

Display();

OflineSave();

Structure();

}

else if(Scommand==6)

{

DeleteAllContacts();

OflineSave();

Structure();

}

}

else

{

throw(Scommand);

}

}

catch(int Scommand)

{

cout<<" You Enter wrong Command... Run the Code Again"<<endl;

Structure();

}

}

};

int main()

{

ContactBook cb;

cb.reopenCB();

string n;

cout<<" What is Your Name: ";

cin>>n;

cout<<"\*\*\*\*\*"<<endl;

cout<<" "<<n<<" WELCOME TO CONTACTBOOK "<<endl;

cout<<"\*\*\*\*\*"<<endl;

cb.Structure();

return 0;

}